A similarity-based community detection method with multiple prototype representation

Kuang Zhou[a,b,*], Arnaud Martin[b], Quan Pan[a]

[a]School of Automation, Northwestern Polytechnical University, Xi'an, Shaanxi 710072, PR China
[b]DRUID, IRISA, University of Rennes 1, Rue E. Branly, 22300 Lannion, France

Abstract

Communities are of great importance for understanding graph structures in social networks. Some existing community detection algorithms use a single prototype to represent each group. In real applications, this may not adequately model the different types of communities and hence limits the clustering performance on social networks. To address this problem, a Similarity-based Multi-Prototype (SMP) community detection approach is proposed in this paper. In SMP, vertices in each community carry various weights to describe their degree of representativeness. This mechanism enables each community to be represented by more than one node. The centrality of nodes is used to calculate prototype weights, while similarity is utilized to guide us to partitioning the graph. Experimental results on computer generated and real-world networks clearly show that SMP performs well for detecting communities. Moreover, the method could provide richer information for the inner structure of the detected communities with the help of prototype weights compared with the existing community detection models.

Keywords: Multiple prototype, node similarity, community detection, prototype weights

1. Introduction

In order to have a better understanding of organizations and functions in real-world networked systems, the community structure in the graph is a primary feature that should be taken into consideration [1]. As a result, community detection, which can extract specific structures from complex networks, has attracted considerable attention crossing many areas from physics, biology, and economics to sociology [2, 3], where systems are often represented as graphs. Generally, a community in a network is a subgraph whose nodes are densely connected within itself but sparsely connected with the rest of the network [4–6].

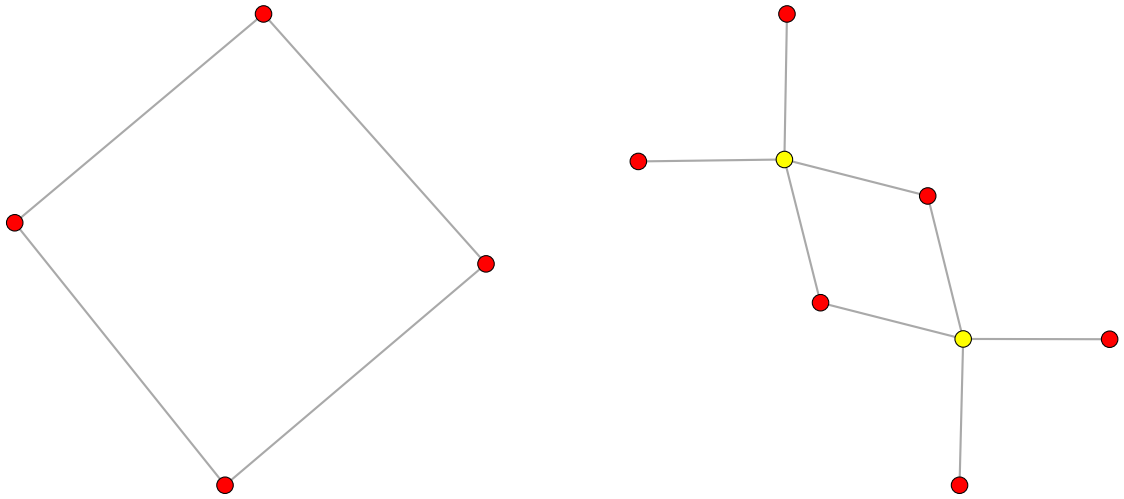
Recently, significant progress has been achieved in this research field and several popular algorithms for community detection have been presented. One of the most popular type of classical methods partitions networks by optimizing some criteria. Newman and Girvan [7] proposed a network modularity measure (usually denoted by Q) and several algorithms that try to maximize Q have been designed [8–10]. But recent researches have found that the modularity based algorithms could not detect communities smaller than a certain size. This problem is famously known as the

*Corresponding author
Email address: kzhoumath@163.com (Kuang Zhou)

resolution limit [11]. The single optimization criteria, *i.e.,* modularity, may not be adequate to represent the structures in complex networks, thus Amiri et al. [12] suggested a new community detection process as a multi-objective optimization problem. Another family of approaches considers hierarchical clustering techniques. It merges or splits clusters according to a topological measure of similarity between the nodes and tries to build a hierarchical tree of partitions [13–18]. Also there are some ways, such as spectral methods [19] and signal process method [6, 20], to map topological relationship of nodes in the graphs into geometrical structures of vectors in n-dimensional Euclidean space, where classical clustering methods like classical C-Means (CM) [6], Fuzzy C-Means (FCM) [5, 20] or Evidential C-Means (ECM) [21] could be evoked. However, there must be some loss of information during the mapping process. Besides, these prototype-based partition methods themselves are sensitive to the initial seeds. For social networks with good community structures, the center of one group is likely to be one person, who plays the leader role in the community. That is to say, one of the members in the group is better to be selected as the seed, rather than the center of all the objects.

To solve these problems, Jiang et al. [6] proposed an efficient algorithm named K-rank which selects the node with the highest centrality value as the prototype. In our previous work, an evidential centrality measure is used to set one "most possible" object in the class to be the prototype [22]. We believe that the characteristic on the prototype of each community is important for community detection. However, in some cases the way of using only one node to describe a community may not be sufficient enough. To illustrate the limitation of one-prototype community representation, we use two simple community structures shown in Figure 1. The first community consists of four members while the second has eight. It can be seen that in the left community, it is unreasonable to describe the cluster structure using any one of the four nodes in the group, since no one of the four nodes could be viewed as a more proper representative than the other three. In the right community in Figure 1, two members (marked yellow) out of the eight are equal reasonable to be selected as the representative of the community. This means choosing any one of them may fail to detect the complete set of all the candidate representative nodes. From these examples, we can see that for some networks, in order to capture various aspects of the community structures, we may need more members rather than one to be referred as the prototypes of an individual group.

Motivated by this idea, in this paper, a Similarity-based Multiple Prototype (SMP) community detection approach is proposed. The centrality values are used as the criterion to select multiple prototypes to characterize each community, and the prototype weights are derived to describe the degree of representativeness of the related objects for their own community. Then the similarity between each node and community is defined, and the nodes are partitioned into divided communities according to these similarities. Here, we emphasize some key points different from those earlier studies and the contribution of this work. Firstly, although there are some multi-prototype clustering methods for the classical data sets [23, 24], there is little such work for community detection problems. Here a new community representation mechanism using multiple prototypes is proposed. Experimental results on artificial and real-world networks show that multiple prototypes are more



a. Community 1 b. Community 2

Figure 1: Two small community's structures. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

powerful than a single center for representing a community, especially for the graphs without clear community structures. Secondly, the concept of prototype weights is presented, which describes the degree of representativeness of a member in its own group. With the help of prototype weights, SMP provides more sufficient description for each individual community. This enables us to gain a deep insight into the internal structure of a community, which we believe is also very important and useful for network analysis. Thirdly, in the proposed community detection approach, different kinds of similarity and centrality measures could be adopted, which makes it more practical and flexible in real applications.

The rest of this paper is organized as follows. In Section 2, some basic concepts and the rationale of our method are briefly introduced. In Section 3, the multi-prototype community detection approach is presented in detail. In order to show the effectiveness of our approach, in Section 4 we test our algorithm on different artificial and real-world networks and make comparisons with the existing methods. Finally, we conclude and present some perspectives in Section 5.

2. Preliminary knowledge

In this section some background knowledge related to community detection problems and social networks, including centrality and similarity measures, modularity and some classical existing algorithms, will be presented.

2.1. Node centrality and similarity

Generally speaking, the person who is the center of a community in a social network has the following characteristics: he has relation with most of the members of the group and the relationships are stronger than usual; he may directly contact with other persons who also play an important role in their own communities. Therefore, the centers of the community should be set to the ones not only with high degree and weight strength, but also with neighbors who also have

high degree and strength. The degree of node is the number of its connections with other nodes, and the strength describes the levels of these connections. Gao et al. [25] proposed an evidential centrality measure, named Evidential Semi-local Centrality (ESC), based on the theory of belief functions. In the application of ESC, the degree and strength of each node are first expressed by basic belief assignments (BBA), and then the fused importance is calculated using the combination rule in the theory of belief functions. The higher the ESC value is, the more important the node is. Gao et al. [25] pointed out that it is more efficient than the existing centrality measures such as Degree Centrality (DC), Betweenness Centrality (BC) and Closeness Centrality (CC). The detail computation process of ESC can be found in [25].

The similarity measures the closeness between any pair of nodes in the graph. In [26] several node similarity metrics on basis of local information were described and the performance of different measures applied to community detection was discussed. Here we give a brief description of some measures. Let $G(V, E)$ be an undirected network, where V is the set of N nodes and E is the sets of m edges. Let $\boldsymbol{A} = (a_{ij})_{N\times N}$ denote the adjacency matrix, where $a_{ij} = 1$ represents that there is an edge between nodes i and j.

(1) Common neighbors. This measure is based on the idea that more common neighbors the pair shares, more similar they are. Thus the similarity can be simply proportional to the number of their shared neighbors:

$$s^{\mathrm{C}}(x, y) = |N(x) \cap N(y)|, \tag{1}$$

where $N(x) = \{w \in V \setminus x : a(w, x) = 1\}$ denotes the set of vertices that are adjacent to x.

(2) Jaccard Index. This index was proposed by Jaccard over a hundred years ago, and is defined as

$$s^{\mathrm{J}}(x, y) = \frac{|N(x) \cap N(y)|}{|N(x) \cup N(y)|}. \tag{2}$$

(3) Zhou-Lü-Zhang Index. Zhou et al. [26] also proposed a new similarity metric which is motivated by the resource allocation process:

$$s^{\mathrm{Z}}(x, y) = \sum_{z\in N(x)\cap N(y)} \frac{1}{d(z)}, \tag{3}$$

where $d(z)$ is the degree of node z.

Pan et al. [27] pointed out that the similarity measure proposed by Zhou et al. [26] may bring about inaccurate results for community detection on the networks as the metric can not differentiate the tightness relation between a pair of nodes whether they are connected directly or indirectly. In order to overcome this defect, in his presented new measure the similarity between unconnected pair is simply set to be 0:

$$S^{P}(x, y) = \begin{cases} \sum\limits_{z\in N(x)\cap N(y)} \frac{1}{d(z)}, & \text{if } x, y \text{ are connected,} \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

A similarity measure considering the global graph structure is put forward by Hu et al. [20] based on signaling propagation in the network. For a network with N nodes, every node is viewed

as an excitable system which can send, receive, and record signals. Initially, a node is selected as the source of signal. Then the source node sends a signal to its neighbors and itself first. Afterwards, the nodes with signals can also send signals to their neighbors and themselves. After a certain T time steps, the amount distribution of signals over the nodes could be viewed as the influence of the source node on the whole network. Naturally, compared with nodes in other communities, the nodes of the same community have more similar influence on the whole network. Therefore, similarities between nodes could be obtained by calculating the differences between the amount of signals they have received.

2.2. Modularity

Recently, many criteria were proposed for evaluating the partition of a network. A widely used measure called modularity, or Q function was presented by Newman and Girvan [7]. Let $G(V, E, W)$ be an undirected network, V is the set of N nodes, E is the set of edges, and $\boldsymbol{W}$ is a $N \times N$ edge weight matrix with elements $w_{ij}, i, j = 1, 2, \cdots, N$. Given a hard partition with K groups $\boldsymbol{U} = (u_{ik})_{N\times K}$, where u_{ik} is one if vertex i ($i = 1, 2, \cdots, N$) belongs to the k_{th} ($k = 1, 2, \cdots, K$) community, 0 otherwise. Denote the K crisp subsets of vertices by $\{C_1, C_2, \cdots, C_K\}$, then the modularity can be defined as [1]:

$$Q_h = \frac{1}{\|\boldsymbol{W}\|} \sum_{k=1}^{K} \sum_{i,j\in C_k} \left(w_{ij} - \frac{k_i k_j}{\|\boldsymbol{W}\|} \right), \tag{5}$$

where $\|\boldsymbol{W}\| = \sum_{i,j=1}^{N} w_{ij}$, $k_i = \sum_{j=1}^{N} w_{ij}$.

The Q measure has been proved highly effective in practice for community evaluation, although Fortunato and Barthelemy [11] claim resolution limits of modularity-based division methods. Besides, some other problems of Newman's modularity have also been found [28]. To solve these problems, some new modularity measures have been proposed [28, 29]. In this paper, the Max–Min (MM) modularity function proposed by Chen et al. [28] is utilized as the index to determine the optimal number of communities. MM modularity attempts to maximize the number of edges within groups and minimize the number of unrelated pairs from the user-defined unrelated pair set within groups at the same time:

$$Q_{\mathrm{MM}} = Q_{\max} - Q_{\min}, \tag{6}$$

where $Q_{\max}$ is the Q modularity of the original graph, while $Q_{\min}$ is that of the complement graph $G^{'}$. Graph $G^{'} = (Y, E^{'})$ is created based on the user-defined criteria $\mathcal{M}$ which defines whether two disconnected nodes i, j are related $(i, j) \in \mathcal{M}$ or unrelated $(i, j) \notin \mathcal{M}$, *i.e.,* $(i, j) \in E^{'}$ if $(i, j) \notin E$ and $(i, j) \notin \mathcal{M}$. The related pairs $\mathcal{M}$ can be given by experts, or defined according to the original structure [28].

2.3. Some classical methods of community detection

In Section 4 we will compare the proposed algorithm with five existing methods: K-rank algorithm [6], Multi-level Modularity Optimization (MMO) algorithm [8], Leading Eigenvector (LE) algorithm [30], Label Propagation (LP) algorithm [31], and Information Map (InfoMap) algorithm [32]. Thus here we give a short presentation of these five approaches.

MMO is a heuristic method based on modularity optimization, and the algorithm is divided into two phases repeated iteratively. In the beginning of the first phase, the network is thought to have N groups each of which consists of only one node. Then for each node i, it may be placed into a new community (it must be a community that one of its neighbors belongs to) for which the gain of modularity is maximum. The first phase is not completed until no further improvement of the modularity can be achieved. The second phase consists in building a new network whose nodes are the communities detected in the last phase, and then the first phase can be reapplied on this newly created graph. Blondel et al. [8] pointed out that MMO outperformed all other known community detection methods in terms of computation time.

Newman [30] demonstrated that the modularity can be succinctly expressed as a function of the eigenvalues and eigenvectors of the modularity matrix and derived a competitive Leading Eigenvector (**LE**) algorithm for identifying communities. The graph is first divided into two groups according to the signs of the elements of the eigenvector corresponding to the most positive eigenvalue of the modularity matrix, and then can be partitioned into more communities depending on the requirement analogously. It is showed that LE works better than the standard spectral partitioning method as it is unconstrained by the need to find groups of any particular size [30] .

LP is investigated by Raghavan et al. [31] and it only uses the network structure and requires neither optimization of a predefined objective function nor prior information about the communities. In this model every node is initialized with a unique label. Afterwards each node adopts the label that most of its neighbors currently have at every step. In this iterative process densely connected groups of nodes form a consensus on a unique label to form communities.

InfoMap uses the probability flow of random walks on a network as a proxy for information flows in the real system, and graph clustering turns then into the coding problem of finding the partition that yields the minimum description length of an infinite random walk [1]. The network is optimally decomposed into modules by compressing the information needed to describe of the process of information diffusion across the graph [32]. The regularities in the community structure and their relationships are reflected by a map.

***K*-rank** algorithm is proposed by Jiang et al. [6], and it uses an alternate iteration strategy like K-means. Firstly, the top–K nodes with the highest rank centrality is selected as initial seeds. This initialization mechanism could overcome the problem brought by the random initial centers in the application of prototype-based clustering methods like K-means. Then the seeds and cluster labels are updated alternately by using an iterative technique. As illustrated before, the way of selecting K representative members with each to totally represent one individual community may be insufficient to fully characterize a community. This in turn indicates that multiple nodes should be utilized in order to capture each group in the network more accurately.

3. The multi-prototype community detection approach

We propose here our method. After an introduction of the concept of representative weights (also called prototype weights) in Section 3.1, the whole algorithm will be presented in detail in Section 3.2. The problem of determining the optimum community number and the complexity of the algorithm will be discussed in Section 3.3 and 3.4 respectively.

3.1. The prototype weights

Suppose $C = \{C_1, C_2, \cdots, C_K\}$ is a partition of a graph $G(V, E)$, where V is the set of nodes and E is the set of edges. The N nodes in the graph can be denoted by $\{n_1, n_2, \cdots, n_N\}$. The matrix $\boldsymbol{V}_{K\times N}$ denotes the prototype weights of N nodes with respect to all the K communities. As analyzed before, the centrality value of a node can be used to express the belief that the node plays the center role in its community. Therefore, the probabilistic weight of node j's degree of representativeness in cluster C_r can be derived as below:

$$V_{rj} = \begin{cases} \frac{P_r(j)}{\sum_{\{h:n_h\in C_r\}} P_r(h)} & n_j \in C_r \\ 0 & n_j \notin C_r, \end{cases} \quad r = 1, 2, \cdots, K,\ j = 1, 2, \cdots N, \tag{7}$$

where $P_r(j)$ is the centrality of node n_j in the subgraph corresponding community C_r. Then, for a given node n_i, the similarity between n_i and community C_j, denoted by $\bar{s}_{ij}$, can be obtained as

$$\bar{s}_{ij} = \sum_{h=1}^{N} v_{jh} s_{ih}, \tag{8}$$

where s_{ih} is the similarity between nodes n_i and n_h. From Eqs. (7) and (8) we can see that $\bar{s}_{ij}$ is a weighted sum of the similarity between node n_i and all the nodes in community C_j, and the weights used in the summation depend on the contribution of the nodes to their own community.

3.2. The detection algorithm

The whole SMP algorithm to detect communities in social networks is summarized as Algorithm 1. In fact SMP is a variation of K-means, K-medoids and K-rank. The difference between SMP and the other three clustering algorithms lies in the manner of updating the prototypes. K-means uses the average value to represent every class while K-medoids and K-rank uses one "most possible" object. On the contrary, SMP adopts an effective multi-prototype representation based on the determined prototype weights of each member in the group. Due to the various types of community structures, the way to represent a cluster using multiple prototypes is more reasonable in real applications. Moreover, SMP often needs fewer iterations than K-means to make the algorithm convergent.

Remark: As we can see, SMP provides us a crisp (hard) partition of the analyzed network. Also the similarity between node n_i and community C_j could be obtained by Eq. (8). Then the node n_i's membership with regard to community C_j can be defined as follows:

$$u_{ij} = \frac{\bar{s}_{ij}}{\sum_{h=1}^{K} \bar{s}_{ih}}, \quad i = 1, 2, \cdots, N,\ j = 1, 2, \cdots, K. \tag{9}$$

This form of membership measure is in line with that got by FCM algorithm, where the membership values assigned to an object are inversely related to the relative distance to the cluster. Similarly here the memberships in Eq. (9) are determined by the relative similarities. One of the problem of fuzzy membership has been reported is that it could not distinguish between "equal evidence" (membership values are large and equal for a number of alternatives) and "ignorance" (all the membership values are equal but very close to zero) [33, 34]. If node n_i is equidistant from more

Algorithm 1 : The Similarity-based Multi-Prototype (SMP) community detection algorithm

Input: K, the number of communities; $\boldsymbol{A}$, the adjacency matrix; $\boldsymbol{W}$, the weight matrix (if any); $N_{\max}$, the maximum number of iterations.

Initialization:

(1). Select the top K nodes with highest centralities as the initial K prototypes.

(2). Calculate the similarity matrix between any two nodes in the graph.

(3). Extract the similarity matrix between the nodes and the prototypes. Partition the node into the community to which its nearest prototype belongs, and get the initial K classes of the graph: $C_1, C_2, \cdots, C_K$.

repeat

(4). Update the matrices $\boldsymbol{V}_{K\times N}$ recording prototype weights of N nodes with respect to all the K communities based on the current partitions using Eq. (7).

(5). Calculate the similarity between node n_i and community C_j, $\bar{s}_{ij}$, using Eq. (8), and then cluster the vertices into k communities with every node being in the community it is most similar to.

until All the detected communities remain unchanged or the number of iterations comes to $N_{\max}$.

Output: The membership of each node and the prototype weights of all the members in each community.

than one community, the membership of each cluster will be the same, regardless of the absolute values of the similarity to the communities. Consequently, the fuzzy membership could not be applied to detect noise objects (outliers) which are far but equidistant to some communities [34]. In SMP, the prototype weights can help us solve this problem, which we will show in detail in Section 4.2.

3.3. Determining the number of communities

In the first step of SMP algorithm, the additional information about the number of communities (K) should be specified. This is also a fundamental issue in classical CM and FCM clusterings. In fact, to determine the optimal number of clusters is an open problem for prototype-based clustering methods. Most of the methods to solve this problem consist in computing a validity index from several community structures detected with different values of K and looking for a minimum or maximum of a given criterion [5, 20, 35]. In this paper MM-modularity (Eq. (6)) is used to estimate a proper K. The modularity values signify the quality of the detected communities. When the modularity achieves the maximum, we can get the best K.

3.4. The complexity of SMP algorithm

The complexity of SMP consists of calculating similarities and centralities of nodes and iterative process. If we use signal similarity and evidential semi-local centrality measures, as we will see in Section 4, the corresponding time complexity if $\mathrm{O}(c(|k|+1)N^2)$ [20] and $\mathrm{O}(N|k|^2)$ [25], where c is the number of propagation, $|k|$ is the average degree of vertices in the network, and N is the number of nodes. The iterative technique is similar to that in K-means. The only difference is

the strategy of updating the prototypes. K-means computes the average value of all the members in the cluster, while SMP tries to find prototype weights of all the members. As the communities are subgraphs which are much smaller than the original network, the updating prototype weights process of SMP does not cost much. If the number of communities K is fixed, the time complexity of K-means clustering is $\mathrm{O}(NKt)$, where t is the number of iterations. Consequently, the total complexity of SMP is $\mathrm{O}(c(|k|+1)N^2 + N|k|^2 + NKt)$. It is worth noting that SMP often needs fewer iterations.

4. Experimental results

In this section some experiments are performed on both computer-generated graphs and real-world networks whose community structure is known in advance. Apart from K-rank [6], we also compare SMP with four other classical methods: Multi-level Modularity Optimization (MMO) algorithm [8], Leading Eigenvector (LE) algorithm [30], Label Propagation (LP) algorithm [31], and Information Map algorithm (InfoMap) [32] presented in Section 2.3. The obtained community structures are evaluated with known performance measures, i.e., accuracy and NMI (Normalized Mutual Information). As the benchmarks and the real-world data sets used in this paper are with known community structure, accuracy and NMI measure the similarity between the planted partitions (ground truth) and the results of the algorithms. The NMI of two partitions A and B of the graph, $I(A,B)$, can be calculated by

$$I(A,B) = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} N_{ij}\log(\frac{N_{ij}n}{N_{i\cdot}N_{\cdot j}})}{\sum_{i=1}^{C_A} N_{i\cdot}\log(\frac{N_{i\cdot}}{n}) + \sum_{j=1}^{C_B} N_{\cdot j}\log(\frac{N_{\cdot j}}{n})}, \tag{10}$$

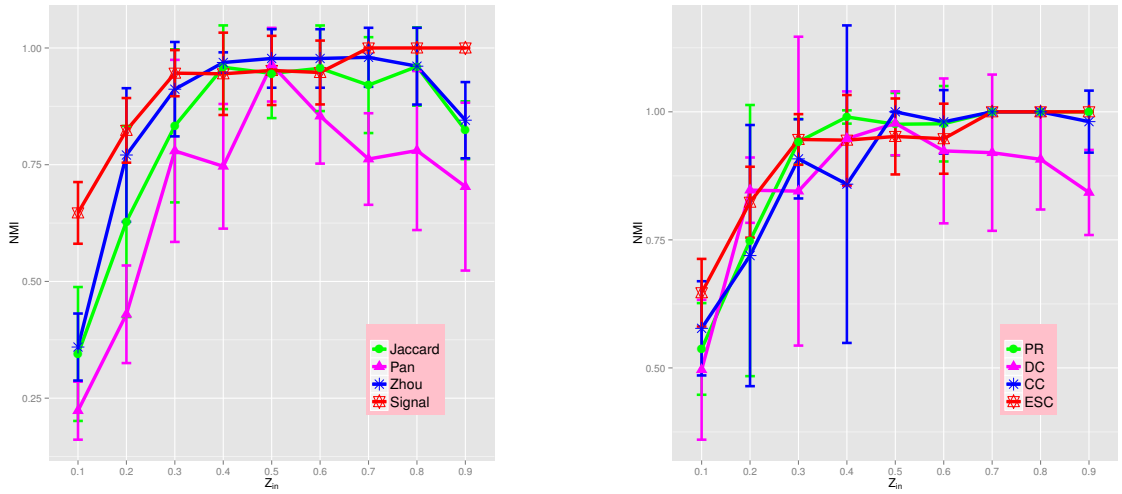
where C_A and C_B denote the numbers of communities in partitions A and B respectively. The notation N_{ij} denotes the element of matrix $(\boldsymbol{N})_{C_A\times C_B}$, representing the number of nodes in the i_{th} community of A that appear in the j_{th} community of B. The sum over row i of matrix $\boldsymbol{N}$ is denoted by $N_{i\cdot}$ and that over column j by $N_{\cdot j}$. Both accuracy and NMI measure the proportion of the nodes that have been grouped correctly, and represent the consistence between the found community structure and the presumed one [20, 36]. The influence of different similarity and centrality measures in the application of SMP will be discussed in the first experiment. After that we will use the evidential semi-local centrality and signal similarity in the following tests based on the experimental results.

4.1. Computer-generated graphs

The algorithm is first compared by means of two classes of computer-generated artificial benchmark networks, namely, Girvan and Newman [3] (GN) and Lancichinetti et al. [37] benchmark (LFR) networks. For the former, each network has $N = 128$ nodes in total and 32 nodes in each of the four divided communities. The average degree of each vertex is set to 16. For a given node, the average number of links to its fellows in the inner community, denoted by Z_{in}, is varied from 8 to 16. The average number of edges between communities, denoted by Z_{out}, is varied from 8 to 0. The larger Z_{in} is, the more apparent community structure the network has.

It is noteworthy that in the application of SMP algorithm, different similarity and centrality measures could be adopted instead of the signal similarity and evidential semi-local centrality

suggested in this paper. When using ESC for calculating the centrality, results by four different similarity metrics, *i.e.,* signal similarity, the simple Jaccard index and the measures proposed by Pan et al. [27] (denoted by Pan in the figure) and Zhou et al. [26] (denoted by Zhou in the figure), are shown in Figure 2-a. As can be seen from the figure, the results by signal similarity are better than the other indices in terms of NMI values. Here we could conclude that global similarity measures like signal similarity are more applicable for SMP than local ones. Figure 2-b depicts the behavior of SMP with difference centrality measures but the same (signal) similarity index. It can be seen that ESC and PR are better among the four measures, *i.e.,* ESC, PageRank (PR) [38], Degree Centrality (DC), and Closeness Centrality (CC). Although there is no significant difference between ESC and PR, the performance of ESC is more stable than PR. This paper is not focusing on the comparison of different similarity and centrality measures, thus in the following experiment we only consider the signal similarity and evidential semi-local centrality.



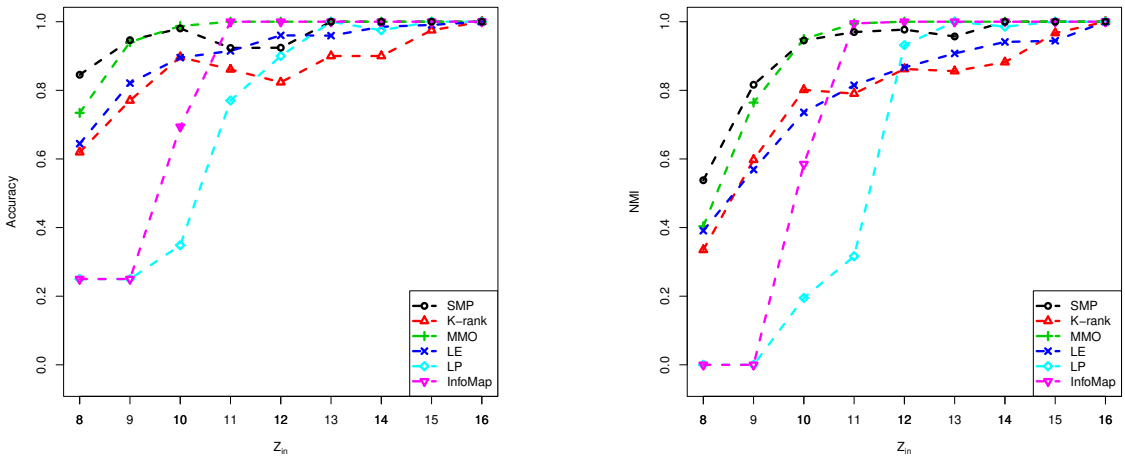
a. Different similarity measures

b. Different centrality measures

Figure 2: Comparison of similarity and centrality measures in the application of SMP algorithm. Average NMI values (plus and minus one standard deviation) for 20 repeated experiments, as a function of the average degree.

For each Z_{in}, the experiment is repeated 20 times and the mean values of the evaluating measures are reported. The average values of the indices by accuracy and NMI using SMP and the other five algorithms with different values of Z_{in} are displayed in Figure 3-a and Figure 3-b respectively. The results show that in terms of accuracy and NMI, all the methods perform well when Z_{in} is large. However, when Z_{in} is smaller than 10, they have different performances. LP and InfoMap have the worst results as they could not work when $Z_{\text{in}} < 10$. SMP and MMO are best in general among all the methods. Although MMO is superior to SMP when $Z_{\text{in}} = 11$ and $Z_{\text{in}} = 12$, the superiority is not obvious. SMP is significantly better than MMO when Z_{in} is small (especially when $Z_{\text{in}} = 8$). Moreover, with the decreasing of Z_{in}, the performance of SMP does not drop so dramatically as the case in other methods. This demonstrates that using multiple members with various prototype weights is able to characterize the structure of clusters more precisely no

matter whether the network has clear community structure or not, which in turn helps to produce a partition of the graph with good quality.



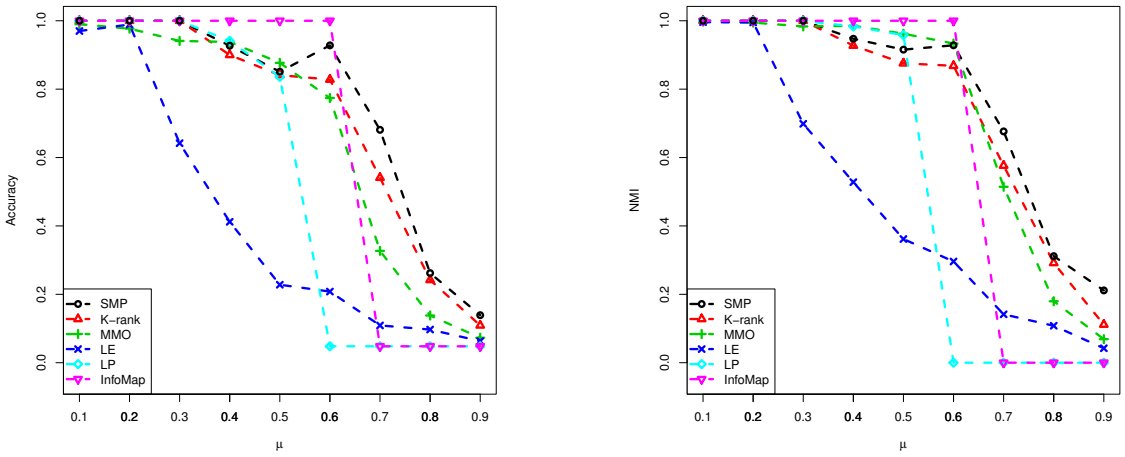
a. Accuracy b. NMI

Figure 3: Comparison of SMP and other algorithms in Girvan and Newman's networks.

The LFR benchmark network [37] is an artificial network for community detection, which is claimed to process some basic statistical properties found in real networks, such as heterogeneous distributions of degree and community size. The results of different methods in three kinds of LFR networks with 1000, 2000 and 5000 nodes are displayed in Figures 4–6 respectively. The parameter μ illustrated in x-axis in the figures identifies whether the network has clear communities. When μ is small, the graph has well community structure. In such a case, almost all the methods perform well. But we can see that when μ is large, the results by SMP have relatively large values of NMI, and the performance of SMP and K-rank do not drop dramatically as the case in other methods. SMP slightly outperforms K-rank especially when μ is large, this could be attributed to the multi-prototype representation of communities. Overall, from the two types of benchmarks, SMP fits for the networks no matter whether they have clear community structures or not.

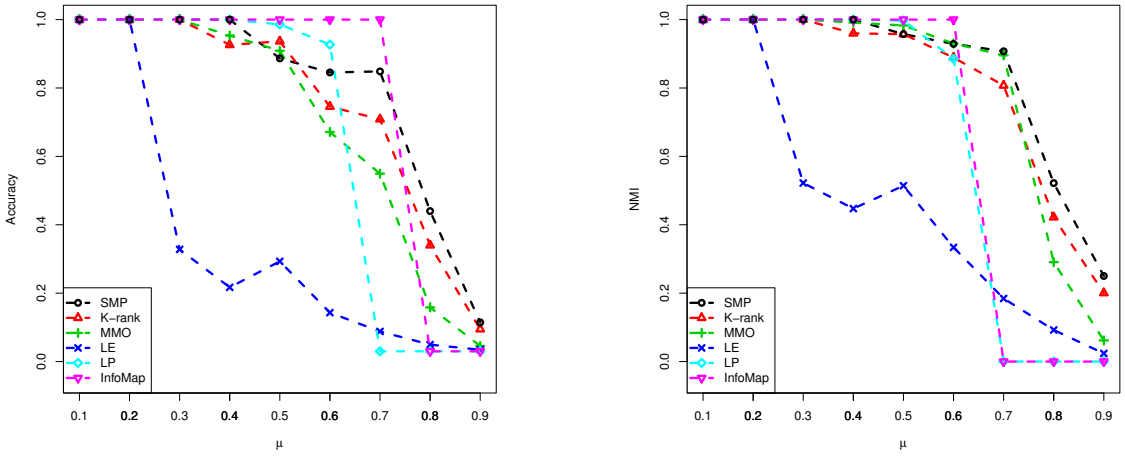
4.2. Real world networks

A. Zachary's Karate Club. To evaluate the effectiveness of the proposed method applied on real-world networks, we first test on a widely used benchmark in detecting community structures, "Karate Club" [39], studied by Wayne Zachary. The network consists of 34 nodes and 78 edges representing the friendship among the members of the club. During the development, a dispute arose between the club's administrator and instructor, which eventually resulted in the club split



a. Accuracy b. NMI

Figure 4: Comparison of SMP and other algorithms in LFR networks. The number of nodes is $N = 1000$. The average degree is $|k| = 20$, and the pair for the exponents is $(\gamma, \beta) = (2, 1)$.

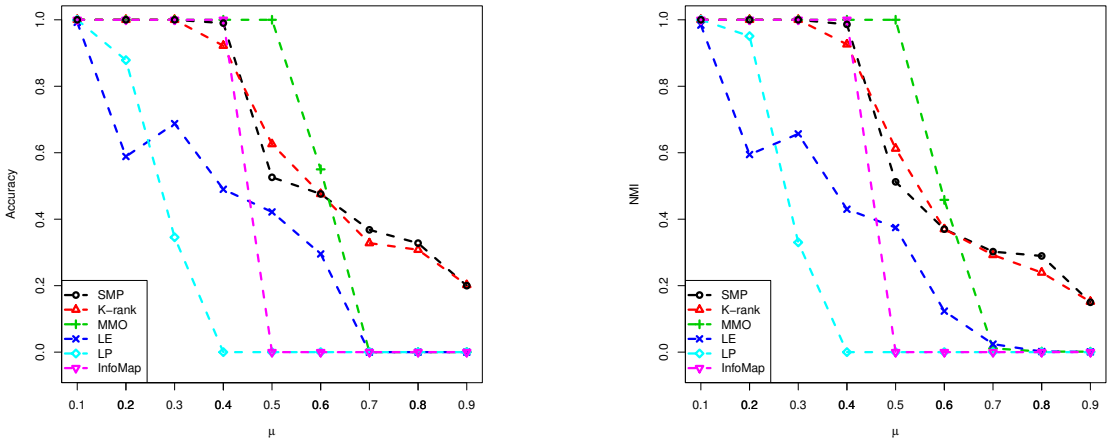


a. Accuracy b. NMI

Figure 5: Comparison of SMP and other algorithms in LFR networks. The number of nodes is $N = 2000$. The average degree is $|k| = 30$, and the pair for the exponents is $(\gamma, \beta) = (2, 1)$.

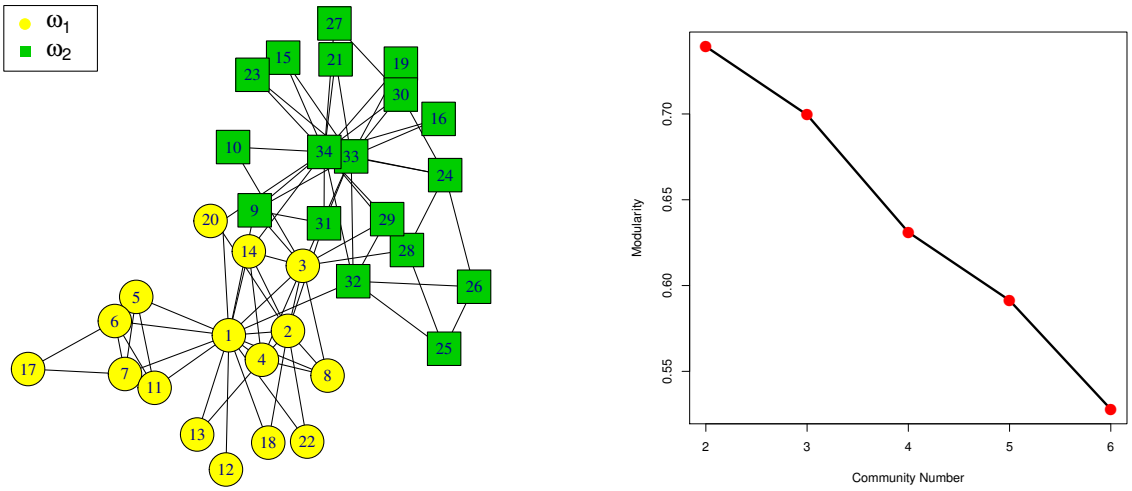
into two smaller clubs, centered around the administrator and the instructor respectively (see Figure 7-a).

The values of the modularity with different number of communities are displayed in Figure 7-b. The modularity function peaks when $K = 2$. This is in consistent with the fact that the network has two groups. The discovered communities are illustrated in Table 1. The table also shows the prototype weights in each of the found group. As we can see, node 1 makes the most contribution to community 1, while node 34 is most important to community 2. This confirms the center role of the two persons in their own communities. On the contrary, nodes 17 and 25 seem not very important in their group in terms of their prototype weights. We can see that in Figure 7-a, these



a. Accuracy b. NMI

Figure 6: Comparison of SMP and other algorithms in LFR networks. The number of nodes is $N = 5000$. The average degree is $|k| = 30$, and the pair for the exponents is $(\gamma, \beta) = (2, 1)$.



a. Original Karate Club network b. Modularity function

Figure 7: The Karate Club network and the modularity values varying with community numbers.

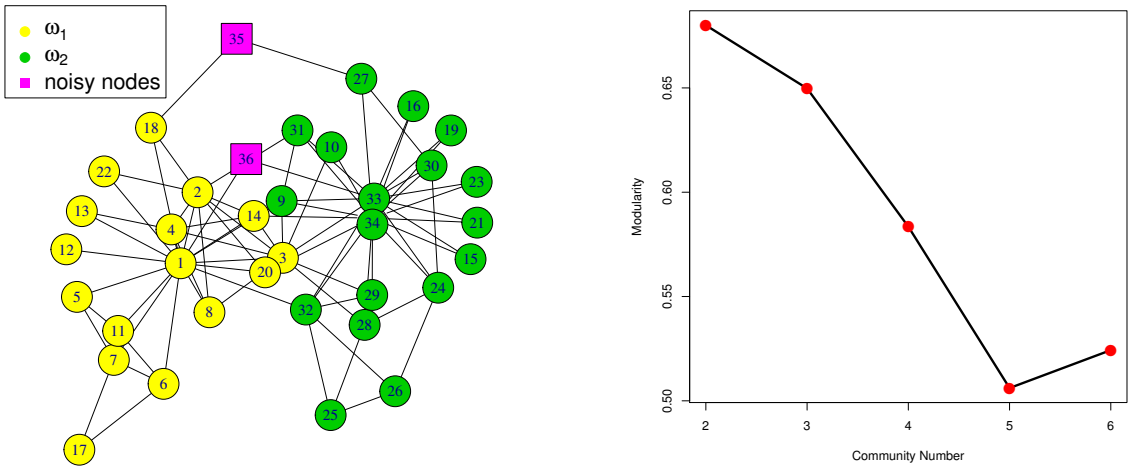
two nodes locate in the marginal parts. Therefore, the proposed SMP detection approach enables us to have a better understanding of the graph structure with the help of prototype weights.

B. Karate Club network with some added noisy nodes. In this test, two noisy nodes are added to the original Karate Club network (see Figure 8-a). The first one is node 35, which is directly connected with nodes 18 and 27. The other one is 36, which is connected to nodes 1 and 33. It can be seen that node 36 has stronger relationships with both communities than node 35. This is due to the fact that the nodes connected to node 36 play leader roles in their own groups, but node 35 contacts with two marginal nodes which have only "small" or insignificant roles in their own groups. The modularity values varying with different community numbers are depicted in Figure 8-b and the detected results are displayed in Table 2.

Table 1: The results for Karate Club network. The notation u_{ij} denotes the fuzzy membership of node n_i to community j, and PW is short for prototype weights. The nodes are order by prototype weights in each community.

Community 1				Community 2			
Node ID	u_{i1}	u_{i2}	PW	Node ID	u_{i1}	u_{i2}	PW
1	0.5324	0.4676	0.1166	34	0.4607	0.5393	0.1025
2	0.5305	0.4695	0.0929	33	0.4582	0.5418	0.0940
4	0.5385	0.4615	0.0881	24	0.4469	0.5531	0.0738
3	0.5091	0.4909	0.0857	32	0.4798	0.5202	0.0698
8	0.5404	0.4596	0.0786	30	0.4424	0.5576	0.0679
14	0.5175	0.4825	0.0786	9	0.4882	0.5118	0.0595
6	0.5576	0.4424	0.0536	31	0.4772	0.5228	0.0595
7	0.5576	0.4424	0.0536	15	0.4464	0.5536	0.0532
18	0.5486	0.4514	0.0524	16	0.4464	0.5536	0.0532
20	0.5109	0.4891	0.0524	19	0.4464	0.5536	0.0532
22	0.5486	0.4514	0.0524	21	0.4464	0.5536	0.0532
5	0.5564	0.4436	0.0488	23	0.4464	0.5536	0.0532
11	0.5564	0.4436	0.0488	28	0.4707	0.5293	0.0474
13	0.5513	0.4487	0.0476	29	0.4788	0.5212	0.0408
12	0.5488	0.4512	0.0334	27	0.4420	0.5580	0.0392
17	0.5734	0.4266	0.0164	10	0.4802	0.5198	0.0307
				26	0.4582	0.5418	0.0268
				25	0.4671	0.5329	0.0223

From Table 2 we can see that the fuzzy membership values of nodes 35 and 36 are almost the same for both communities (approximatively equal to 0.5). These results could not reflect the difference between ignorance and uncertainty. As node 35 is only related to one outward node of each community, thus we are ignorant about which community it really belongs to, or we say node 35 is an outlier. On the contrary, node 36 connects with the key members (playing an important role in the community) in both communities. Thus there is uncertainty rather than ignorance about which community node 36 is in. In this network, node 36 is a "good" member for both communities, whereas node 35 is a "poor" member. As mentioned before, the inability to distinguish the outliers from the uncertain nodes with equal memberships is caused by the relative similarity used in fuzzy memberships. In SMP, the prototype weights could be utilized to solve this problem and to detect the outliers. As shown in Table 2, the prototype weight of node 35 is the least in the community, but node 36 contributes much more than node 35. Therefore, node 35 has no contribution to both communities (the prototype weight of node 35 for community 1 is 0.0052, and 0 for community 2), and it could be recognized as an outlier. This example further demonstrates the fact that prototype weights indeed enable us to gain a better understanding of the graph structure, especially for detecting outliers in the network.



a. Karate Club network with two added nodes

b. Modularity function

Figure 8: The Karate Club network with added nodes and the modularity values varying with community numbers.

We also test our method on four other real-world graphs: American football network, Dolphins network, Lesmis network and Political books network[1]. The values of the two indices, accuracy and NMI, applied to evaluate the performance of different methods are listed in Table 3 and Table 4 respectively[2]. It can been seen from the tables, SMP application results in a community structure with highest accuracy level in most cases. In terms of the performance measure NMI, SMP also outperforms the other algorithms. It should be noted that some methods provide partitions with high accuracy but low NMI. This may be caused by the fact that they cluster the nodes into too many small communities. The partition rules of both K-rank and SMP are based on node similarity. These two approaches are better than the others in general, and the effectiveness could be attributed to the high performance of vertex similarities. But the reason that SMP works better than K-rank in these real-world networks is largely because of the application of multiple prototype representation of communities.

From the above extensive experimental results, we can summarize the compelling properties of SMP as follows:

1) In the partition process, SMP uses multiple prototypes to represent the communities. This is a useful extension of the existing community detection methods where only one prototype is allowed, especially when the analyzed graph has some complex community structures.
2) The prototype weights, as a by-product of the detection results, provide us with some valuable information about the community structure from another point of view, and enable us to gain a better understanding of the analyzed graph.
3) SMP works well even for the graphs without clear community structures. It could avoid the problem of inability to distinguish the outliers from uncertain data for fuzzy membership.
4) Last but not the least, the experiments on both synthetic and real-world graph data sets demon-

[1]These data sets can be found in http://networkdata.ics.uci.edu/index.php

[2]All these real-world graphs are with known community structure, thus the accuracy and NMI are calculated based on the ground truth and the partition got by different algorithms.

Table 2: The results for Karate Club network with added nodes. The notation u_{ij} denotes the fuzzy membership of node n_i to community j, and PW is short for prototype weights. The nodes are order by PW in each community.

Community 1				Community 2			
Node ID	u_{i1}	u_{i2}	PW	Node ID	u_{i1}	u_{i2}	PW
1	0.5278	0.4722	0.1111	34	0.4656	0.5344	0.1028
2	0.5271	0.4729	0.0888	33	0.4651	0.5349	0.0944
4	0.5344	0.4656	0.0836	24	0.4534	0.5466	0.0737
3	0.5084	0.4916	0.0814	32	0.4824	0.5176	0.0696
8	0.5360	0.4640	0.0747	30	0.4506	0.5494	0.0680
14	0.5158	0.4842	0.0747	9	0.4899	0.5101	0.0598
18	0.5399	0.4601	0.0528	31	0.4801	0.5199	0.0598
6	0.5511	0.4489	0.0520	15	0.4533	0.5467	0.0534
7	0.5511	0.4489	0.0520	16	0.4533	0.5467	0.0534
20	0.5099	0.4901	0.0506	19	0.4533	0.5467	0.0534
22	0.5427	0.4573	0.0506	21	0.4533	0.5467	0.0534
5	0.5498	0.4502	0.0475	23	0.4533	0.5467	0.0534
11	0.5498	0.4502	0.0475	28	0.4740	0.5260	0.0471
13	0.5454	0.4546	0.0462	29	0.4813	0.5187	0.0404
12	0.5427	0.4573	0.0330	27	0.4539	0.5461	0.0395
36	0.5016	0.4984	0.0330	10	0.4826	0.5174	0.0309
17	0.5658	0.4342	0.0154	26	0.4628	0.5372	0.0258
35	0.5020	0.4980	0.0052	25	0.4705	0.5295	0.0212

strate that the proposed approach is a competitive candidate for community detection tasks compared with other five existing methods.

Table 3: Comparison of SMP and other algorithms by accuracy in real-world networks.

	Karate	Football	Dolphins	Lesmis	Books
SMP	**1.0000**	**0.9345**	**1.0000**	0.7792	**0.8667**
K-rank	**1.0000**	0.9320	**1.0000**	0.8052	0.8537
MMO	**1.0000**	0.8000	0.9516	0.7922	0.7276
LE	**1.0000**	0.6261	0.9677	0.7273	0.8476
LP	0.9706	0.9043	**1.0000**	0.7273	0.8476
InfoMap	**1.0000**	0.9043	0.9839	**0.8701**	0.7854

5. Conclusion

In this paper, a new type of similarity-based community detection algorithm called SMP is proposed. SMP could find not only communities of each node but also weighted representative members of each group. In real world community detection problems, information on both community labels and internal structure of each of the detected communities are important. One

Table 4: Comparison of SMP and other algorithms by NMI in real-world networks.

	Karate	Football	Dolphins	Lesmis	Books
SMP	**1.0000**	**0.9235**	**1.0000**	0.7444	**0.5938**
K-rank	**1.0000**	0.9211	**1.0000**	0.7818	0.5741
MMO	0.6873	0.8550	0.4617	0.7551	0.5121
LE	0.6552	0.6952	0.5094	0.7182	0.5201
LP	0.8255	0.9095	0.8230	0.7381	0.5485
InfoMap	0.8255	0.8937	0.5629	**0.8198**	0.4935

distinctive characteristic of the proposed method is that each community is presented by multiple prototypes, rather than by single one object. The experiments on synthetic networks show the effectiveness of the proposed method and the tests on real-world networks have further pointed out our method preforms better than the existing ones. The results show that the way of using prototype weights to represent a cluster enables SMP to capture the various types of community structures more precisely and completely hence improves the quality of the detected communities. Moreover, more detail information on the discovered clusters may be obtained with the help of prototype weights.

In real applications, the signal similarity measure and ESC centrality utilized in the work could be replaced by any other index. For instance, if we want to apply the method to directed networks, the similarity and centrality measures for directed networks could be adopted. Therefore, we intend to study on the comparison of difference measures and on the application into directed networks in our future research work. Meanwhile, not only centrality but also more other factors should be considered for determining the prototype weights. Hence the way to optimize the prototype weights using the available information as much as possible will also be included in our further study.

Acknowledgements

The authors are grateful to the anonymous reviewers for all their remarks which helped us to clarify and improve the quality of this paper. This work was supported by the National Natural Science Foundation of China (Nos.61135001, 61403310). The study of the first author in France was supported by the China Scholarship Council.

References

[1] S. Fortunato, Community detection in graphs, Physics Reports 486 (3) (2010) 75–174.

[2] L. d. F. Costa, O. N. Oliveira Jr, G. Travieso, F. A. Rodrigues, P. R. Villas Boas, L. Antiqueira, M. P. Viana, L. E. Correa Rocha, Analyzing and modeling real-world phenomena with complex networks: a survey of applications, Advances in Physics 60 (3) (2011) 329–412.

[3] M. Girvan, M. E. Newman, Community structure in social and biological networks, Proceedings of the National Academy of Sciences 99 (12) (2002) 7821–7826.

[4] M. E. Newman, Modularity and community structure in networks, Proceedings of the National Academy of Sciences 103 (23) (2006) 8577–8582.

[5] S. Zhang, R.-S. Wang, X.-S. Zhang, Identification of overlapping community structure in complex networks using fuzzy c-means clustering, Physica A: Statistical Mechanics and its Applications 374 (1) (2007) 483–490.

[6] Y. Jiang, C. Jia, J. Yu, An efficient community detection method based on rank centrality, Physica A: Statistical Mechanics and its Applications 392 (9) (2013) 2182–2194.

[7] M. E. Newman, M. Girvan, Finding and evaluating community structure in networks, Physical review E 69 (2) (2004) 026113.

[8] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, Journal of Statistical Mechanics: Theory and Experiment 2008 (10) (2008) P10008.

[9] A. Clauset, M. E. Newman, C. Moore, Finding community structure in very large networks, Physical review E 70 (6) (2004) 066111.

[10] J. Duch, A. Arenas, Community detection in complex networks using extremal optimization, Physical review E 72 (2) (2005) 027104.

[11] S. Fortunato, M. Barthelemy, Resolution limit in community detection, Proceedings of the National Academy of Sciences 104 (1) (2007) 36–41.

[12] B. Amiri, L. Hossain, J. W. Crawford, R. T. Wigand, Community Detection in Complex Networks: Multi–objective Enhanced Firefly Algorithm, Knowledge-Based Systems 46 (2013) 1–11.

[13] A. Lancichinetti, S. Fortunato, J. Kertész, Detecting the overlapping and hierarchical community structure in complex networks, New Journal of Physics 11 (3) (2009) 033015.

[14] J. Huang, H. Sun, J. Han, B. Feng, Density-based shrinkage for revealing hierarchical and overlapping community structure in networks, Physica A: Statistical Mechanics and its Applications 390 (11) (2011) 2160–2171.

[15] B. Yang, J. Di, J. Liu, D. Liu, Hierarchical community detection with applications to real-world network analysis, Data & Knowledge Engineering 83 (2013) 20–38.

[16] J. Kim, T. Wilhelm, Spanning tree separation reveals community structure in networks, Physical Review E 87 (3) (2013) 032816.

[17] Z. Zhang, Z. Wang, Mining overlapping and hierarchical communities in complex networks, Physica A: Statistical Mechanics and its Applications 421 (2015) 25–33.

[18] P. Kim, S. Kim, Detecting overlapping and hierarchical communities in complex network using interaction-based edge clustering, Physica A: Statistical Mechanics and its Applications 417 (2015) 46–56.

[19] S. Smyth, S. White, A spectral clustering approach to finding communities in graphs, in: Proceedings of the 5th SIAM International Conference on Data Mining, 76–84, 2005.

[20] Y. Hu, M. Li, P. Zhang, Y. Fan, Z. Di, Community detection by signaling on complex networks, Physical Review E 78 (1) (2008) 016115.

[21] K. Zhou, A. Martin, Q. Pan, Evidential Communities for Complex Networks, in: Information Processing and Management of Uncertainty in Knowledge-Based Systems, Springer, 557–566, 2014.

[22] K. Zhou, A. Martin, Q. Pan, Z.-g. Liu, Median evidential c-means algorithm and its application to community detection, Knowledge-Based Systems 74 (2015) 69–88.

[23] M. Liu, X. Jiang, A. C. Kot, A multi-prototype clustering algorithm, Pattern Recognition 42 (5) (2009) 689–698.

[24] Y. Wang, L. Chen, J.-P. Mei, Incremental Fuzzy Clustering With Multiple Medoids for Large Data, Fuzzy Systems, IEEE Transactions on 22 (6) (2014) 1557–1568.

[25] C. Gao, D. Wei, Y. Hu, S. Mahadevan, Y. Deng, A modified evidential methodology of identifying influential nodes in weighted networks, Physica A: Statistical Mechanics and its Applications 392 (21) (2013) 5490–5500.

[26] T. Zhou, L. Lü, Y.-C. Zhang, Predicting missing links via local information, The European Physical Journal B 71 (4) (2009) 623–630.

[27] Y. Pan, D.-H. Li, J.-G. Liu, J.-Z. Liang, Detecting community structure in complex networks via node similarity, Physica A: Statistical Mechanics and its Applications 389 (14) (2010) 2849–2857.

[28] J. Chen, O. R. Zaïane, R. Goebel, Detecting Communities in Social Networks Using Max-Min Modularity., in: SDM, vol. 3, SIAM, 20–24, 2009.

[29] J. Scripps, P.-N. Tan, A.-H. Esfahanian, Exploration of link structure and community-based node roles in network analysis, in: Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on, IEEE, 649–654, 2007.

[30] M. E. Newman, Finding community structure in networks using the eigenvectors of matrices, Physical review E 74 (3) (2006) 036104.

[31] U. N. Raghavan, R. Albert, S. Kumara, Near linear time algorithm to detect community structures in large-scale networks, Physical Review E 76 (3) (2007) 036106.

[32] M. Rosvall, C. T. Bergstrom, Maps of random walks on complex networks reveal community structure, Proceedings of the National Academy of Sciences 105 (4) (2008) 1118–1123.

[33] R. Krishnapuram, J. M. Keller, A possibilistic approach to clustering, Fuzzy Systems, IEEE Transactions on 1 (2) (1993) 98–110.

[34] N. R. Pal, K. Pal, J. M. Keller, J. C. Bezdek, A possibilistic fuzzy c-means clustering algorithm, Fuzzy Systems, IEEE Transactions on 13 (4) (2005) 517–530.

[35] T. Nepusz, A. Petróczi, L. Négyessy, F. Bazsó, Fuzzy communities and the concept of bridgeness in complex networks, Physical Review E 77 (1) (2008) 016107.

[36] Y. Fan, M. Li, P. Zhang, J. Wu, Z. Di, Accuracy and precision of methods for community identification in weighted networks, Physica A: Statistical Mechanics and its Applications 377 (1) (2007) 363–372.

[37] A. Lancichinetti, S. Fortunato, F. Radicchi, Benchmark graphs for testing community detection algorithms, Physical Review E 78 (4) (2008) 046110.

[38] S. Brin, L. Page, The anatomy of a large-scale hypertextual Web search engine, Computer networks and ISDN systems 30 (1) (1998) 107–117.

[39] W. W. Zachary, An information flow model for conflict and fission in small groups, Journal of anthropological research (1977) 452–473.